# Technological Analysis

## Requirements

### Performance

As the Optimal Camera Placement problem is NP-Hard (Meaning non-deterministic non-polynomial complexity), the project needs to run as fast as possible so that the solution to the problem can be achieved in a reasonable amount of time.

Whilst compiled languages are considered faster than interpreted ones, it is possible to achieve the same or even better performance with an interpreter. An example being Forth which is an interpreted language faster than C which is compiled. As such, more importance should be assigned towards writing optimal algorithms rather than picking an optimised technology (As without sufficient knowledge, writing in a faster technology can lead to slower code).

### Platform Support

As security companies tend to work directly at clients’ locations, running on portable devices such as laptops and tablets is crucial. It is not expected for the program to be executed on phones due to the small screen compared to the needed precision for the planning of Optimal Camera Placement.

As such, certain technologies would need to be excluded. For example, the C# application supports different .NET versions on Windows, Android and macOS. This would require either use the oldest common version (Which would decrease performance/quality of life for Windows users) or use a different version for each platform which would be difficult to maintain.

### Human Readable

The program would consist of several complicated algorithms, each one of them using the level of mathematics not normally taught at university-level Computer Science. High human readability would allow simplifying the debugging process, allowing extra time to be spent on more features or optimisation.

In general, Python is considered the most human-readable language currently available. In contrast, low-level languages such as C or C++ generally are less human-readable due to the requirement for memory control or typecasting.

### Independent

If possible, the program should run without requiring additional software or any technical knowledge. The best-case scenario is a program either being a web application or an executable file as most users have enough technical knowledge to conduct these operations.

This would exclude Python, at least in its basic form, as it requires an additional interpreter to be installed. Java also is not ideal as it is necessary to install JVM to run the code.

## Evaluation

These requirements severely limit the number of technologies that can be used for this project. The main candidates would be executables, such as programming languages from the .NET family or Web applications.

In this case, I have decided to settle on a Web Application due to its accessibility and requirement for nothing more than an internet connection. It also has the advantage of putting some of the program’s load on the client as interface generation would be handled by the browser. This leaves the server with more processing power to calculate optimal camera placement problem.

An additional advantage of this approach is the usage of the main server, which in general is significantly more powerful than an average computer.

The main disadvantage is the requirement for an internet connection. In most cases security companies have their own network, using a slow but stable satellite connection. This allows them to operate independently across the world.

For the actual Framework, I have decided to go with Flask. Flask is a minimalistic Python web framework, with comparable performance to Spring Boot or .NET Core. This effectively means that users would receive their requests as soon as they are completed.

Another advantage of Python is that whilst the language itself is slow, many core libraries, such as Numpy are written in C. This gives a large performance boost whilst still being in a readable form.